1

Data Cleaning

Drop & Keep

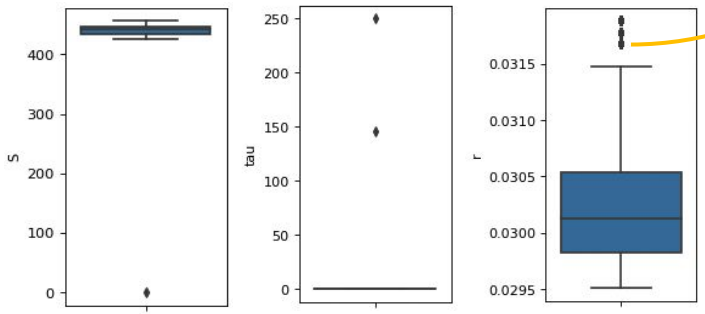
Null Values

- There are 5 missing values in 2 rows
- Considering the sample size, it is safe to remove all 2 rows

	Value	S	K	tau	r	BS
292	8.625	NaN	NaN	NaN	0.03003	Over
818	NaN	431.284616	NaN	0.230159	0.02972	Over

Outliers

- There are 3 outliers in S and T
- There are several outliers in r



2

Feature Engineering

Overview

Start from 4

We have four basic independent variables in our dataset.

- S
- K
- r
- T

3 Methods

We view the data from 3 different aspects to construct relevant features.

- Profitability
- Future & Present Values
- Bins

19 Variables

We have 19 variables with business significance in total.

\ Hoooooray! /

Feature Details

Method	Name	Formula	Comments
Profitability	K-S Ratio	K/S	The increase of the asset value
	K-S Difference	K-S	
	K-S Ratio per Day	K/(S*τ)	The break even increase of the asset value per day
Future & Present Values	Risk-free FV	S*(1+r)^τ	The risk free future value of the asset (based on Current Asset Value)
	Risk-free FV Difference	K-Risk-free FV	The difference between Risk-free FV and K
	Risk-free FV Proportion	(K-(Risk-free FV))/K	The proportion of difference between Risk-free FV and K to K
	Asset Interest Rate	(K/S)^(1/τ)-1	The theoretical interest rate of the asset
	Asset Interest Rate Difference	Asset Interest Rate - r	The difference between Asset Interest Rate and r
	Asset Interest Rate Proportion	(Asset Interest Rate-r)/Asset Interest Rate	The proportion of difference between Asset Interest Rate and r
	Risk-free PV	K/(1+r)^τ	The risk free present value of the asset (based on Strike Price of Option)
Bins	Quintile of S/ Quintile of K/ Quintile of r	\	Approximately equally bin data into 5 parts
	extremely low K	K < 404	54 out of 1675 observations
	extremely high r	r > 0.031	274 out of 1675 observations

3

Feature Selection

Forward Selection

SequentialFeatureSelector function

Regression: scoring = 'r2'

add variables in this order	variable name
1	risk_free_gap_abs
2	tau
3	risk_free_rate_prop
4	risk_free_FV
5	K
6	KS_tau_ratio
7	r_asset
8	K_bin
9	is_high_r
10	r_bin
11	r
12	S
13	is_low_K
14	KS_diff
15	risk_free_gap_prop
16	KS_ratio
17	S_bin
18	risk_free_rate_abs
19	S_expected

Classification: scoring = 'accuracy'

add variables in this order	variable name
1	KS_ratio
2	risk_free_FV
3	r_bin
4	is_low_K
5	S_bin
6	KS_tau_ratio
7	S_expected
8	r
9	K_bin
10	r_asset
11	risk_free_gap_abs
12	K
13	risk_free_rate_abs
14	is_high_r
15	KS_diff
16	tau
17	S
18	risk_free_rate_prop
19	risk_free_gap_prop

4

Model Exploration and Selection

Linear Models:

Regression: Linear Regression

Classification: Logistic Regression

Prediction Accuracy

Nonlinear Models:

Random Forest

Elastic Net

Neural Network

Support Vector Machine

K-Nearest Neighbors

Light Gradient Boosting Machine

eXtreme Gradient Boosting

Model Exploration

1

Split training data

70% → training
30% → testing

2

Training set

Include different numbers of variables: 3, 5, 10, 15, 19

3

Training set

5-fold cross-validation
GridSearchCV function → best parameter settings

4

Testing set

Predict the values or labels
Calculate out-of-sample R^2 and accuracy

Model Performance (Regression)

- **Linear Regression**: # of variables: 19
  - **R^2 in training: 0.9929, R^2 in testing: 0.9932**
- **Random Forest**: # of variables: 10; Best set of parameters: {'max_depth': 25, 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 90}
  - **R^2 in training: 0.9984, R^2 in testing: 0.9989**
- **Elastic Net**: # of variables: 19; Best set of parameters: {'alpha': 0.0001, 'l1_ratio': 0.99}
  - **R^2 in training: 0.9895, R^2 in testing: 0.9906**
- **Neural Network**: # of variables: 19; Best set of parameters: {'activation': 'logistic', 'alpha': 0.001, 'hidden_layer_sizes': (30, 30), 'learning_rate': 'adaptive', 'max_iter': 3000, 'solver': 'adam'}
  - **R^2 in training: 0.9987, R^2 in testing: 0.9987**
- **Epsilon-Support Vector Regression**: # of variables: 3; Best set of parameters: {'kernel': 'rbf', 'gamma': 'auto', 'C': 10000, 'epsilon': 0.01, 'shrinking': 'True'}
  - **R^2 in training: 0.9977, R^2 in testing: 0.9992**
- **K-Nearest Neighbors**: # of variables: 5; Best set of parameters: {'leaf_size': 1, 'n_neighbors': 8, 'p': 1}
  - **R^2 in training: 0.9898, R^2 in testing: 0.5448**
- **Light Gradient Boosting Machine**: # of variables: 19; Best set of parameters: {'learning_rate': 0.1, 'max_depth': 20, 'min_child_samples': 10, 'n_estimators': 200}
  - **R^2 in training: 0.9985, R^2 in testing: 09991**
- **eXtreme Gradient Boosting**: # of variables: 5; Best set of parameters: {'learning_rate': 0.1, 'max_depth': 5, 'min_child_weight': 5, 'n_estimators': 500, 'nthread': 4, 'objective': 'reg: squarederror'}
  - **R^2 in training: 0.9989, R^2 in testing: 0.9993**

Model Performance (Classification)

- **Logistic Regression**: # of variables: 10
  - **R^2 in training: 0.9206, R^2 in testing: 0.9165**
- **Random Forest**: # of variables: 10; Best set of parameters: {'max_depth': 25, 'min_samples_leaf': 4, 'min_samples_split': 5, 'n_estimators': 10}
  - **R^2 in training: 0.9275, R^2 in testing: 0.9324**
- **Elastic Net**: # of variables: 15; Best set of parameters: {'alpha': 0.00065, 'l1_ratio': 0.73, 'learning_rate': 'optimal', 'penalty': 'elasticnet'}
  - **R^2 in training: 0.9198, R^2 in testing: 0.9066**
- **Neural Network**: # of variables: 10; Best set of parameters: {'activation': 'logistic', 'alpha': 0.01, 'hidden_layer_sizes': (30, 30), 'learning_rate': 'constant', 'max_iter': 2000, 'solver': 'adam'}
  - **R^2 in training: 0.9232, R^2 in testing: 0.9105**
- **Support Vector Machine**: # of variables: 10; Best set of parameters: {'C': 1, 'kernel': 'rbf', 'gamma': scale, 'shrinking': 'False', probability: 'False'}
  - **R^2 in training: 0.9322, R^2 in testing: 0.9264**
- **K-Nearest Neighbors**: # of variables: 19; Best set of parameters: {'leaf_size': 1, 'n_neighbors': 14, 'p': 1}
  - **R^2 in training: 0.9318, R^2 in testing: 0.5714**
- **Light Gradient Boosting Machine**: # of variables: 10; Best set of parameters: {'learning_rate': 0.2, 'max_depth': 20, 'min_child_samples': 20, 'n_estimators': 50}
  - **R^2 in training: 0.9292, R^2 in testing: 0.9364**
- **eXtreme Gradient Boosting**: # of variables: 15; Best set of parameters: {'eval_metric': 'error', 'learning_rate': 0.1, 'max_depth': 5, 'min_child_weight': 5, 'n_estimators': 100, 'nthread': 4, 'objective': 'binary:logistic'}
  - **R^2 in training: 0.9335, R^2 in testing: 0.9404**

Final Model

eXtreme Gradient Boosting (XGBoost):

- Highest out-of-sample R^2
- Highest accuracy (lowest classification error)

Regression:

Number of variables: 5

Best set of parameters: {'learning_rate': 0.1, 'max_depth': 5, 'min_child_weight': 5, 'n_estimators': 500, 'nthread': 4, 'objective': 'reg:squarederror'}

R^2 in training: 99.89%

R^2 in testing: 99.93%

Classification:

Number of variables: 15

Best set of parameters: {'eval_metric': 'error', 'learning_rate': 0.1, 'max_depth': 5, 'min_child_weight': 5, 'n_estimators': 100, 'nthread': 4, 'objective': 'binary:logistic'}

Accuracy in training: 93.35%

Accuracy in testing: 94.04%

6

Summary

Summary

1

Data Cleaning

Checked outliers and missing values
Dropped rows with missing values and extreme outliers

2

Feature Engineering

Created 15 new variables using the given four variables

3

Feature Selection

Applied the forward stepwise selection method

4

Model Exploration

Tried different algorithms and tuned models with different hyperparameters

7

Limitations and Future Steps

Limitations & Future Steps

Limitation

- Limited data on political, social, economic events
- Lack of domain expertise during feature engineering
- Limited computational power

Current model

Future Steps

- Gather more data
- Seek domain experts' advice
- Upgrade our computational power to improve the tuning process
- Use different feature selection methods